Responsive Data Table Solution with New Scrolling Control Gesture for Better User Experience

Lukáš Čegan[1]

[1] University of Pardubice, Faculty of Electrical Engineering and Informatics, Department of Information Technology, Pardubice, Czech Republic, lukas.cegan@upce.cz

Abstract. According to available statistics, nowadays more user browse websites on mobile devices than on a desktop computer. This brings new challenges for web developers on how to deal with such a heterogeneous environment of devices. Today, several different solutions are on the market for building responsive data tables. However, these solutions have their limits in the event that the table contains a large number of records. In this paper a new solution is proposed based on a new scroll control gesture and elimination of superfluous loading records. The impacts of the proposed solution to the user experience were investigated through an experiment whose results are presented and discussed in this paper.

Keywords: Data table; User Experience; Responsive design

1 Introduction

Dynamic development in the mobile technology area, cheaper and more sophisticated devices on the market, faster and achievable communications network, all bring an increasing number of users who access the Internet via mobile devices. According to related statistics, the number of mobile users was greater than desktop users. It brings to web developers a variety of new challenges which deal with design and programming web applications. These applications must be designed for such a large group of diverse devices, while providing a good user experience. Currently, one of the most used approaches to deal with this fact is the application of responsive design.

The term "Responsive web design" was coined by a leading web developer Ethan Marcotte in his book [1]. He describes responsive design as a new way of designing the ever-changing web. The goal of responsive design is to create easy web reading and content navigation with minimal resizing, scrolling and panning, across a wide range of different devices from smart phones to desktop computer monitors. Also, the responsive web design is a subject of numerous scientific studies today. A variety of approaches have been proposed to implement responsive design in mobile e-learning systems [2], [3] or support better mobile access to virtual and remote laboratories [4]. An additional published solution is focused on adaptation of usability principles in responsive web design technique in the e-commerce area [5]. Also, some authors present a solution for responsive web design based on the resolution [6]. All of these

studies have only one goal to achieve a better user experience which is nowadays the holy grail of all web developers. Responsive design is just one of the techniques to achieve a better user experience. In addition, there are many other techniques to improve user experience. For example, some authors present in their papers solutions how to improve the user experience on mobile apps through data mining, based on collecting user context data by Google Services API [7]. Or, the authors try to enrich user experience by visualizing video sounds with sound word animation [8].

This paper is focused on improving the user experience with a web data table by combine responsive design and a new way of controlling and loading table records. The data table is often the dominant component of websites because it offers the ability to display a lot of information in a standard structured way, which every user understands well. Unfortunately, making data tables mobile friendly, is in a large number of cases like trying to cram an entire basketball team into a Volkswagen Beetle car. Very often it is impossible. If the developer does not apply any responsive design technique, on the small screen, the data table may be displayed in two ways. First, the table can be reduced to the size of the screen (see Fig. 1 A). This leads to the fact that the user must use the microscope to be able to read the table values. Second, the table can remain in regular size but at a cost that the user sees on the screen, only a small portion of the table at a time (see Fig. 1 B). In this case, the user is able to read the values of the table, but to get an overview of the entire table, the user must repeatedly move the contents of a table on the screen, which is very time-consuming and annoying.

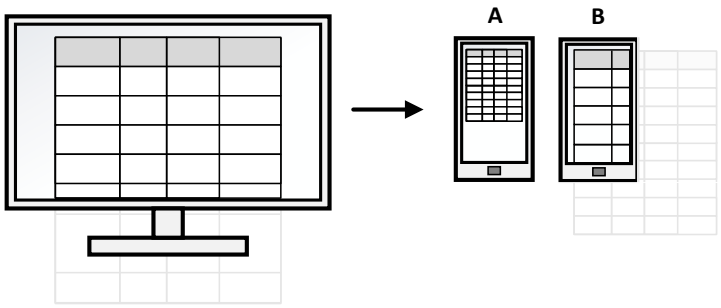

Fig. 1. Data table – desktop computer / mobile device

The web developers used many different approaches which are used in the practical world for solving this problem. Every approach has a positive and negative impact for user experiences. Unfortunately, there is no single universal solution. It is very important to rigorously assess all factors of table (like the number of rows and columns, format of entries, editable and sortable records, etc.) for selecting an appropriate approach for concrete data table. In this paper a new solution is proposed based on an intelligent data table control and elimination of superfluous loading table records which save the transmitted data and allow the users to more quickly find the records of interest in the data table.

The paper is organized as follows. After introducing the objective of this paper, the responsive design is presented in Section II. The Section III described the proposed solution for viewing multicolumn data table. The section IV presented the practical experiments. The results of the experimental analysis are discussed in section V. Finally, the last section gives conclusions and future research opportunities followed by references at the end.

2 Responsive Data Table

Today, web developers use several different approaches which have been developed for establishing tables that can scale well in different viewport sizes. Common solutions are based on pure HTML and CSS. A suitable example can be a solution implemented in one of the most widespread CSS framework Bootstrap. To create a responsive table, Boostrap applies the overflow-x property: auto to wrap the table and overflow-y property: hidden for a window less than 768px. This causes, on the small screen, the content of table to be available by scrolling horizontally (see next source code).

Example of a Twitter Boostrap 3.0 responsive table solution [9]

```
program Inflation (Output)
<style>
.table-responsive {
  min-height: .01%;
  overflow-x: auto;
}
@media screen and (max-width: 767px) {
  .table-responsive {
    width: 100%;
    margin-bottom: 15px;
    overflow-y: hidden;
    -ms-overflow-style: -ms-autohiding-scrollbar;
    border: 1px solid #ddd;
  }
}
</style>
<div style=”table-responsive”>
  <table>
    ...
  </table>
</div>
```

Besides pure HTML and CSS solutions, advanced approaches based on JavaScript exist also. The example of this approach is responsive a data table from the Foundation Zurb framework. Their solution consists in pinning the first column and

making the rest of the table horizontally scrollable. To achieve this functionality, Zurb uses jQuery library to manipulate the DOM and CSS [10]. However, this is not the only solution on the market. Generally, current responsive data table solutions based on JavaScript can be divided to three basic groups; key/value table, table with hiding less important columns, table with inverted axes.

2.1 Key/Value Data Table

A large data table with many columns cannot be reduced to the size of a small smartphone screen. An approach called Key/Value table consists of replacing each row of the table by its own new table. This new table contains only two columns. The first column contains the "keys", which are the headers of the original table. The second column contains the "values", which are the records of the original row (see Fig. 2). In other words, each row of the source table is displayed in the new small table, which is formed by the first row (header table) plus the actual row of the source table. Then, this table is displayed vertically instead of horizontally. This solution eliminates the need for horizontal scrolling. On the other hand, this solution requires more vertical scrolling. Another disadvantage is the lost possibility to simply compare each value between rows (records no longer positioned in one column).

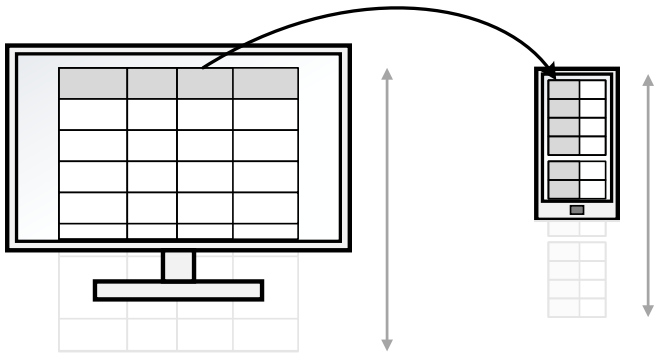

Fig. 2. Key/Value data table

2.2 Table with Hiding Less Important Columns

A solution called "Table with hiding less important columns" is based on predetermined rules that determine which columns will be displayed in the available width of the screen. Each column of the table has a defined priority level that determines the rendering order of the columns. If the columns of the table have no determined level priority, then the columns with the smallest width are rendered as first (see Fig 3). For the user perspective, the biggest disadvantage is loss of control over what part of the content will be displayed. Some of the solutions solve this

problem by adding a drop-down menu through which the user can select columns to display.

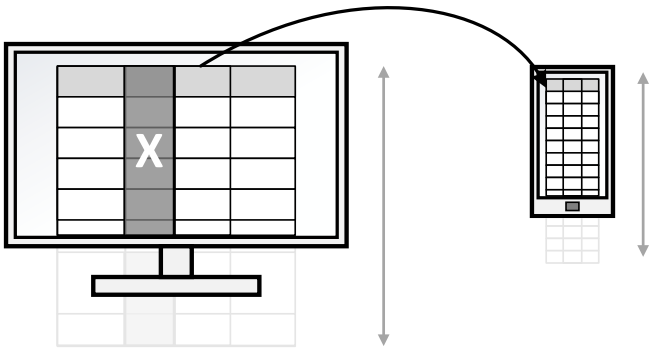

Fig. 3. Table with hiding less important columns

2.3 Table with Inverted Axes

The last solution “Table with inverted axes” is based on the assumption that users used a mobile device to read more in portrait than landscape orientation (see Fig 4). Therefore, if we flip the table axes, we can view more columns at once. In this solution the data table is scrollable in the direction from left to right and vice versa. Due to the user losing an overview of the table, it is necessary to fixed a header of table and enable scrolling only the table entries exclude table header.

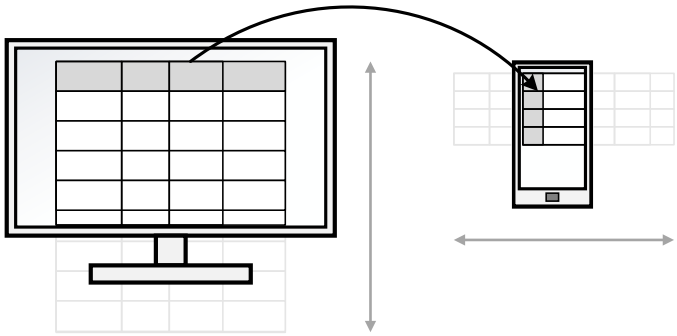

Fig. 4. Table with inverted axes

2.4 Limitation of Responsive Data Table Solutions

Approaches outlined above can help improve user experience in a mobile environment greatly. However, these approaches have their limits in the event that the table contains a large number of records. In this case the records delivered to a user has to be limited. A Commonly used approach is pagination. This means that the control bar is attached to the table from which the users can select a page of records. This method of data control has been successfully used on desktops but in a mobile environment, leads to a worse user experience, because web users have to work intensively with the control buttons. A newer method is sequential loading of records, automatically triggered by scrolling the contents of the table. The disadvantage of this method is a large amount of data that is unnecessarily loaded in the case that the user requests the records at the end of the table.

3 Proposed Solution

Based on the identification of the main limitations of current responsive data table solutions, which are listed in Chapter 2, we proposed a new robust solution that significantly speeds up the web users' ability to achieve the required records, and it eliminates an unnecessary load of records. The proposed solution aims to improve the user experience, which is essential for the successful website.

3.1 Touch Gesture

The core of the proposed solution is based on the new touch gestures through which a mobile user calls up a hint window with the table rows index (see box with letter F in Fig.5). The gesture is detected by a proprietary JavaScript library when the user scrolls the content of the table very quickly (on the background is permanently calculated current speed scrolling). When web browser captures this gesture, normal scrolling of table content is interrupted and instead of it, the user scrolls the index value in a hint window. If the user is satisfied with the selected value, the user closes the selection of the index value by tapping on the hint window. Then web browser hides the hint window, creates an AJAX request and sends it to the web server. Web server processes the request and sends the requested records back to the web browser. Then the user can continue browsing the table from a place that is very close to the target record which the user wants to display.

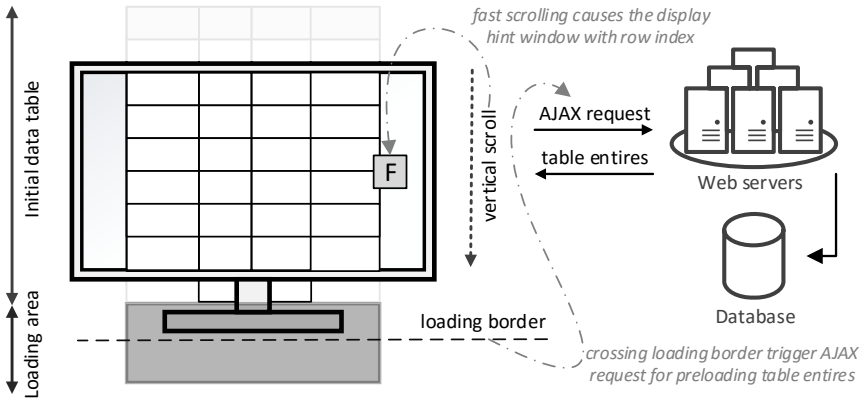

Fig. 5. Architecture of proposed solution

3.2 Loading and Rendering Table Records

The loading of table records is, in the proposed solution, triggered in two different ways. First, the data loading is triggered when the user confirms the selection of an index value as described above. Second, when a user scrolls the table at a normal speed and at the same time, the loading area crossed the loading border (see Fig.5). This event also triggers an AJAX request to the web server as in the first case. Because the records loading consumes a certain amount of time, it is recommended to set the loading border at a distance corresponding to half the height of the device screen in order to achieve continuous data rendering.

The records are included to the table by JavaScript immediately after the web browser receives them from web server. In this case, "include" means to create for each row of records a new DOM elements TR, and TD, and inject them into the right place in the table.

4 Experiment

The effectiveness of the proposed solution has been investigated in three test scenarios. For each scenario a special website was created that has been tested on twenty users of different age and gender. For each scenario, respondents were given a different task. The data for each website comes from OpenFlights service (11). Furthermore, each scenario was tested in various network environments that simulate different bandwidth and latency. To simulate the network environment Linux tools Netem (Network Emulator) and TBF (Token Bucket Filter) was used. The parameters for each scenario was shown in Table 1.

Table 1. Parameters for each experiment scenario

	Scenario A	Scenario B	Scenario C
Dataset	airports.dat – 12 columns, 8107 rows	airlines.dat – 8 columns, 6048 rows	airlines.dat – 8 columns, 6048 rows
Index column	name of airport – first letter	name of airline – first letter	airline ID – multiples of 500
User task	find row for "Ozamis" airport	find row for "Transilvania" airline	find first row for airline ID "11726"
Network	3G – 1 Mbit/s bandwidth, 300ms latency	LTE – 10 Mbit/s bandwidth, 50ms latency	FIBER – unlimited Mbit/s bandwidth, 50ms latency

The performance metrics in this experiment were the total time required to complete the task (TTT), overall size of the transferred data from the server to the client (TDS) and total number of AJAX requests (TNR).

5 Experimental Results and Discussion

For evaluation of efficiency of the proposed solution the method of A/B testing has been used. This method consists in comparing two variants, A and B, which are examined separately in a controlled experiment. Experiment results are statistically analyzed to determine which variation performs better for a given conversion goal.

In our experiment, variant B is a pure HTML table with responsive design "Table with hiding less important columns" (see chapter 2.2) and sequential loading of records. In the variant A same table is used as in variant B, but for records loading the proposed solution is used. The next Table 2 shows the results of an experiment for each test scenario. Values represent medians of all measurements.

Table 2. Results of experiment (medians of all measurement)

	Scenario A		Scenario B		Scenario C	
	A	B	A	B	A	B
Total task time [s]	12	52	14	83	28	325
Transferred data size [kb]	13	196	21	209	19	305
Total number of requests	5	59	7	71	6	129

As the results show, from the user's perspective the proposed solution is significantly better tool than traditional solutions, because it enables users to more quickly reach the searched table records. Especially in networks with high latency is the difference substantial. The proposed solution also significantly reduces the amount of transferred data which could be reflected in a reduction of costs paid to the ISP. Both of these characteristics are certainly a positive impact on the user experience. For a better

comprehensibility the experiment results are also shown in the graphs (see Fig. 6). In each graph both variant A/B are shown. Each variant is composed of three bars expressing each scenario.

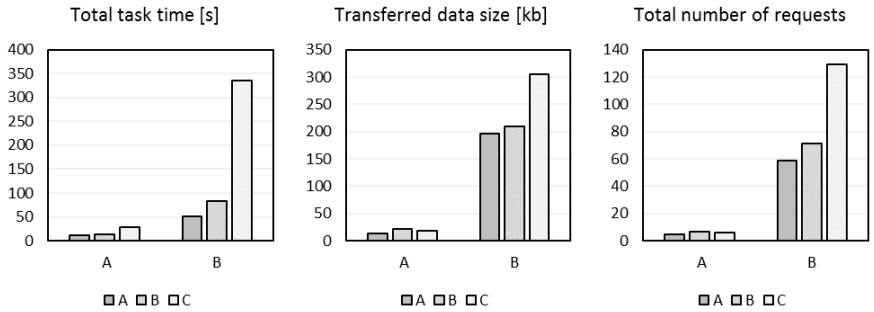

Fig. 6. Comparison of experiment results – 1) Total task time, 2) Transferred data size, 3) Total number of requests

6 Conclusion

This paper presents a new approach to display and control a data table in a web environment. The proposed solution is based on the combination of responsive design and a new control gesture through which a user can work faster with table records. Moreover, the proposed solution also significantly reduces amount of transferred data. Both of these improvements lead to better user experience. The effectiveness of the proposed solution has been investigated in a controlled experiment on the three test scenarios. The experiment results are discussed in this paper and confirm the initial research assumptions.

As future work, we are planning to implement new module to our solution which allows the web user to simply select a column of a table under which will be realized, fast scrolling of the records. In the current version of the solution is the selection of a column only in the hands of developers, which is a significant limitation of functionality from the user's perspective.

Acknowledgments. This work is published thanks to the financial support Faculty of Electrical Engineering and Informatics, University of Pardubice under grant TG02010058 “Podpora aktivit proof-of-concept na Univerzitě Pardubice”.

References

1. Marcotte, E.: Responsive web design. Second Ed. A book apart (2014)
2. Zhu, B.: Responsive Design: e-Learning Site Transformation. In: Fourth International Conference on Networking and Distributed Computing, pp. 126--130. IEEE (2013)
3. Peng, W., Zhou, Y.: The Design and Research of Responsive Web Supporting Mobile Learning Devices: e-Learning Site Transformation. In: International Symposium on Educational Technology, pp. 163--167. IEEE (2015)
4. Zervas, P., Trichos, A., Sampson D.G., Li, N." A Responsive Design Approach for Supporting Mobile Access to Virtual and Remote Laboratories: e-Learning Site Transformation. In: 14th International Conference on Advanced Learning Technologies, pp. 11--13. IEEE (2014)
5. Majid, E.S.A., Kamaruddin N., Mansor, Z.: Adaptation of usability principles in responsive web design technique for e-commerce development. In: International Conference on Electrical Engineering and Informatics, pp. 726--729. IEEE (2015)
6. Lee, J., Lee I., Kwon, I., Yun, H., Lee, J., Jung, M., Kim, H.: Responsive Web Design According to the Resolution. In: 8th International Conference on u- and e-Service, Science and Technology, pp. 1--5. IEEE (2015)
7. Auad, T.O.S., Mendes L.F.C., Stroele, V., David, J.M.N.: Improving the user experience on mobile apps through data mining. In: 20th International Conference on Computer Supported Cooperative Work in Design, pp. 158--163. IEEE (2016)
8. Wang, F., Nagano, H., Kashino, K., Igarashi, K.: Visualizing Video Sounds with Sound Word Animation to Enrich User Experience. IEEE Transactions on Multimedia, Issue 99, IEEE (2016)
9. Twitter Boostrap – Overview, http://getbootstrap.com/css/#tables-responsive
10. Responsive Tables. ZURB Foundation, http://foundation.zurb.com/responsive-tables.html
11. Airport, airline and route data, http://openflights.org/data.html